

Name:

Instructions

1. Write your name at the top of *this* page.
2. This is a closed-book exam. No form of communication is permitted (eg, talking, texting, etc.), except with the course staff.
3. No electronic devices are permitted.
4. There are 30 multiple-choice/short-answer questions in this exam, each worth 3 points. You have 75 minutes to answer the questions.
5. Each question must be answered *with a pencil* as shown below. It will be marked as incorrect otherwise.

Multiple-choice question: (A) (B) (C) (D) (E)

Short answer question:

6. You may use the blank spaces for any scratch work.
7. Discussing the exam contents with anyone who has not taken the exam is a violation of the academic honesty code.

Problem 1. Consider the following table, which gives the running time $T(n)$ in seconds for a program for various values of the input size n :

n	$T(n)$
1000	5
2000	40
4000	320
8000	2560

a. What is the value of $T(n)$ if $n = 16000$?

b. What is the running time classification for the program?

- (A) Linearithmic
- (B) Cubic
- (C) Quadratic
- (D) Quartic
- (E) Linear

Problem 2. What is the running time classification for following code fragment?

```
int sum = 0;
for (int i = 0; i < n; i++) {
    for (int j = 0; j < 1000; j++) {
        for (int k = 0; k < n; k++) {
            sum++;
        }
    }
}
```

- (A) Cubic
- (B) Linearithmic
- (C) Exponential
- (D) Quadratic
- (E) Linear

Problem 3. Consider a data type τ with three instance variables: `int x`, `long y`, and `double z`. Ignoring array and object overheads, what is the memory footprint (in kilobytes) of the array `a[]` created and initialized as follows?

```
T[] a = new T[1000];
for (int i = 0; i < 1000; i++) {
    T[i] = new T();
}
```

Problem 4. Consider the following recursive functions:

```
public static int f(Node x) {
    return (x == null) ? 0 : 1 + f(x.next);
}

public static int g(Node x) {
    return (x == null) ? 0 : x.item * x.item + g(x.next);
}
```

- a. What does $f(a)$ return, where a is a reference to the first node in the linked list containing the items 1, 2, 3, 4, 5, 6, and 7 and in that order?
- b. What does $g(a)$ return, where a is a reference to the first node in the linked list containing the items 1, 2, 3, 4, 5, 6, and 7 and in that order?

Problem 5. Consider the following functions:

```

public static Iterator<Character> f(String s) {
    Queue<Character> Q = new Queue<Character>();
    for (int i = 0; i < s.length(); i++) {
        if (i % 2 != 0) {
            Q.enqueue(s.charAt(i));
        }
    }
    return Q.iterator();
}

public static Iterator<Character> g(String s) {
    Stack<Character> S = new Stack<Character>();
    for (int i = 0; i < s.length(); i++) {
        if (i % 2 == 0) {
            S.push(s.charAt(i));
        }
    }
    return S.iterator();
}

```

a. What is the value returned by `f("alice").next()`?

b. What is the value returned by the method call `g("alice").next()`?

Problem 6. Suppose we use the `QuickFindUF` data structure to solve the dynamic connectivity problem with 10 sites and input pairs (8, 1), (7, 6), (9, 2), (7, 8), (4, 6), (6, 3), and (4, 1), arriving in that order; the code for the `union()` method in `QuickFindUF` is shown below.

```

public void union(int p, int q) {
    int pID = find(p);
    int qID = find(q);
    if (pID == qID) {
        return;
    }
    for (int i = 0; i < id.length; i++) {
        if (id[i] == pID) {
            id[i] = qID;
        }
    }
    count--;
}

```

a. How many components are left after all the pairs are processed?

b. What is the size of the largest component?

c. What is the identifier of the largest component?

Problem 7. Consider sorting an array `a[]` containing the following strings, using selection sort (shown below):

C H P V T Q N M J W

```

public static void sort(Comparable[] a) {
    int n = a.length;
    for (int i = 0; i < n; i++) {
        int min = i;
        for (int j = i + 1; j < n; j++) {
            if (less(a[j], a[min])) {
                min = j;
            }
        }
        exchange(a, i, min);
    }
}

```

a. What is the value that `p` is exchanged with?

b. What is the value that `v` is exchanged with?

Problem 8. Consider sorting an array `a[]` containing the following strings, using insertion sort (shown below):

C H J M P Q L V T W

```

public static void sort(Comparable[] a) {
    int n = a.length;
    for (int i = 1; i < n; i++) {
        for (int j = i; j > 0 && less(a[j], a[j - 1]); j--) {
            exchange(a, j, j - 1);
        }
    }
}

```

Where is the item `L` sorted (ie, what is its index) relative to the items before it?

Problem 9. Consider sorting an array `a[]` containing the following strings (already shuffled), using quick sort (shown below):

N C M U F X V G W P

```

public static void sort(Comparable[] a) {
    StdRandom.shuffle(a);
    sort(a, 0, a.length - 1);
}

private static void sort(Comparable[] a, int lo, int hi) {
    if (hi <= lo) {
        return;
    }
    int j = partition(a, lo, hi);
    sort(a, lo, j - 1);
    sort(a, j + 1, hi);
}

private static int partition(Comparable[] a, int lo, int hi) {
    int i = lo;
    int j = hi + 1;
    Comparable v = a[lo];
    while (true) {
        while (less(a[++i], v)) {
            if (i == hi) {
                break;
            }
        }
        while (less(v, a[--j])) {
            if (j == lo) {
                break;
            }
        }
        if (i >= j) {
            break;
        }
        exchange(a, i, j);
    }
    exchange(a, lo, j);
    return j;
}

```

a. What is the destination index of the pivot element after the first call to `partition()`?

b. What is the state of the array `a` after the first call to `partition()`?

- (A) C F G M N P U V W X
 (B) X W V U P N G M C F
 (C) F C M G N X V U W P
 (D) F C M G N P U V W X
 (E) C F G M N X V U W P

c. What is pivot element in the next call to `partition()`?

Problem 10. Insert the following keys in that order into a min-heap:

H Z R B S Y G V O

a. What is the index of the key `Y`?

b. What is the key with index 8?

c. If we perform a `delMin()` operation on the tree, what is the key that will replace the current minimum before it is sunk down?

Problem 11. Consider inserting the following key-value pairs in that order into a symbol table `st`.

key:	C	A	X	E	U	A	W	B	L	O	W	R
value:	1	2	3	4	5	6	7	8	9	10	11	12

a. What is the value returned by `st.size()`?

b. What is the value returned by `st.get("W")`?

Problem 12. Consider inserting the following keys (assume values to be non `null` and arbitrary) into a binary search tree (BST) symbol table `st`, an object of type `BST`.

J S R I X C T D G E W L

a. What is the height of the BST (assume root to be at height 0)?

b. What is the value returned by `st.floor("M")`?

c. What is the value returned by `st.rank("M")`?

d. What is the order in which the keys are visited if we traverse the BST in pre-order?

- (A) J I C D G E S R X L T W
- (B) C D E G I J L R S T W X
- (C) J I C D G E S R L X T W
- (D) J S R I X C T D G E W L
- (E) E G D C I L R W T X S J

e. What is the order in which the keys are visited if we traverse the BST in in-order?

- (A) E G D C I L R W T X S J
- (B) J S R I X C T D G E W L
- (C) J I C D G E S R L X T W
- (D) C D E G I J L R S T W X
- (E) C D E G I J L R S T X W

f. What is the order in which the keys are visited if we traverse the BST in post-order?

- (A) E G D C I L R W T X S J
- (B) J S R I X C T D G E W L
- (C) E G D C I L R W X T S J

- Ⓓ J I C D G E S R L X T W
 Ⓔ C D E G I J L R S T W X

Problem 13. Consider inserting the following keys (assume values to be non null and arbitrary) into an initially empty hash table of $M = 5$ lists, using separate chaining. Use the hash function $h(k) = k \bmod M$ to transform the k th letter of the alphabet¹ into a table index, where $1 \leq k \leq 26$.

J S R I X C T D G E W L

a. What is the length of the longest chain?

b. Which of the following keys is in the longest chain?

- Ⓐ L
 Ⓑ c
 Ⓒ T
 Ⓓ w
 Ⓔ x

¹Letters in the English alphabet along with corresponding values of k : A-1, B-2, C-3, D-4, E-5, F-6, G-7, H-8, I-9, J-10, K-11, L-12, M-13, N-14, O-15, P-16, Q-17, R-18, S-19, T-20, U-21, V-22, W-23, X-24, Y-25, Z-26

Solution 1.

- a. 20480
- b. B

Solution 2. D**Solution 3.** 20**Solution 4.**

- a. 7
- b. 140

Solution 5.

- a. 'l'
- b. 'e'

Solution 6.

- a. 4
- b. 6
- c. 3

Solution 7.

- a. "J"
- b. "M"

Solution 8. 3**Solution 9.**

- a. 4
- b. C
- c. "F"

Solution 10.

- a. 6
- b. "Z"
- c. "V"

Solution 11.

- a. 10
- b. 11

Solution 12.

- a. 5
- b. "L"
- c. 7
- d. C
- e. D
- f. A

Solution 13.

- a. 4
- b. E